John von Neumann Institute for Computing

Towards Shape Optimization for Ventricular Assist Devices Using Parallel Stabilized FEM

M. Behbahani, M. Behr, M. Nicolai, M. Probst

published in

NIC Symposium 2008,
G. Münster, D. Wolf, M. Kremer (Editors),
John von Neumann Institute for Computing, Jülich,
NIC Series, Vol. **39**, ISBN 978-3-9810843-5-1, pp. 325-332, 2008.

http://www.fz-juelich.de/nic-series/volume39

Towards Shape Optimization for Ventricular Assist Devices Using Parallel Stabilized FEM

Mehdi Behbahani, Marek Behr, Mike Nicolai, and Markus Probst

Chair for Computational Analysis of Technical Systems (CATS)
Center for Computational Engineering Science (CCES)
RWTH Aachen University
Steinbachstr. 53B, 52074 Aachen, Germany
E-mail: {mb, behr, nicolai, probst}@cats.rwth-aachen.de
http://www.cats.rwth-aachen.de

Over the last decade, computer simulations of fluid flow have steadily gained acceptance as an effective tool for evaluation of design modifications. The flow features in complex geometries such as blood pumps, as well as the evolution of those features resulting from design changes, are hard to predict even by experienced design engineers. Linking a suitable mathematical framework with appropriate models in order to evaluate shape modifications will ultimately allow not only to analyze flow features but also to compute an optimal design directly. Such a framework requires an immense amount of computing power, which makes it a perfect candidate to exploit the potential of parallel processing on high-performance systems such as the Jülich Blue Gene/L.

1 Introduction

Diseases of the heart are a leading cause of death in the industrialized nations. The most reliable therapy for end-stage heart failure—heart replacement via a transplant—can be applied only in a fraction of the cases because of dramatic shortage of suitable donor hearts. Since 1960's, attempts are being made to design a mechanical solution to heart failure; such a solution can take the form of a full replica of the heart—dual pumping chambers and complex valves—or, more commonly, of an assisting device, which pumps the blood from the existing failing ventricle into the aorta. The latter is referred to as Ventricular Assist Device, or VAD.

The Chair for Computational Analysis of Technical Systems (CATS) at the RWTH Aachen University, under the direction of Prof. M. Behr, is specializing in Computational Fluid Dynamics (CFD) analysis and has been working on simulation of blood flow in VADs since 2000, with the latest analyses focusing on the miniature MicroMed DeBakey VAD, see Fig. 1. The DeBakey pump is an axial pump that provides a countinuous flow. Its design renders it suitable for both long-term and pediatric use.

2 Simulation

The computational flow analyses described here are performed with XNS, an in-house CFD code for simulations of unsteady fluid flows in situations involving significant deformations of the computational domain. The underlying equations are the incompressible Navier-Stokes equations. Although solving these equations analytically remains out of reach, there has been considerable progress in the development of numerical methods that can provide

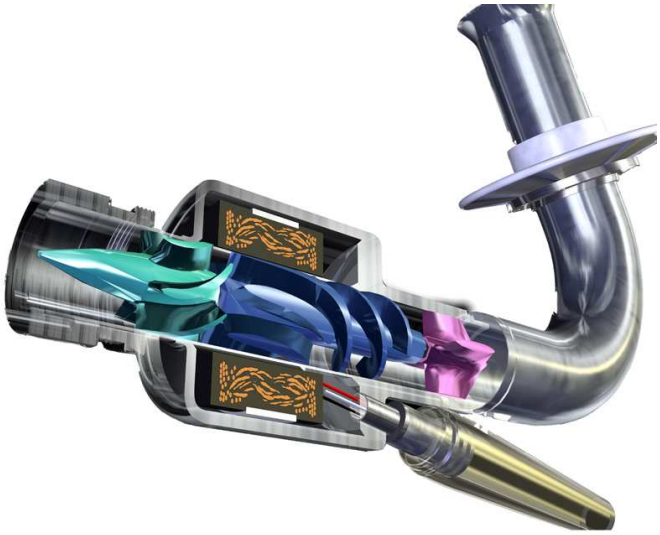


Figure 1. DeBakey VAD geometry. From right to left: inflow cannula, straightener, impeller and diffuser.

reliable approximations of the solution. XNS uses stabilized finite element (FE) formulations together with a space-time discretization on unstructured three-dimensional meshes. Important features for simulating the flow in a VAD include the ability to handle rapidly translating or rotating boundaries using the shear-slip mesh update method (SSMUM), as well as simulating flows of microstructured (in particular viscoelastic) liquids.

Simply computing the flow field inside the VAD presents a formidable challenge regarding the required computing power. Resolving the flow characteristics appropriately necessitates a high-resolution mesh with millions of elements, see Fig. 4. The use of hundreds or even thousands of processors is inevitable; the pump's impeller operates at speeds up to 10,000 rpm necessitating simulations that span thousands of time steps.

In the following sections, we report on two crucial aspects in the design of VADs that can be investigated by numerical simulations. As the most basic requirement, the pump has to be able to operate against a certain pressure head to provide a desired flow rate of 2–5 l/min. Another concern is biocompatibility after implantation; the flow patterns should not induce blood cell damage (hemolysis) above a critical level where it can cause multiple-organ failure, and recirculation zones should be minimized to prevent blood clotting (thrombosis).

Hydraulic Performance

The function of a blood pump is to provide the human body with the required blood flow volume of typically 5 l/min for an adult. Simulations aim to predict the performance of a pump under operating conditions. In order to confirm numerically obtained results it is possible to compare them to measured values from in-vitro experiments, i.e., experiments in a closed mock-up flow loop using an original blood pump.

From a point of view of pump performance it is important to determine the pressure

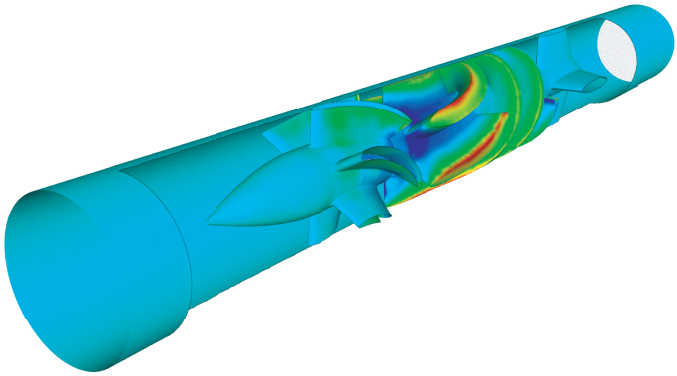


Figure 2. Sample pressure distribution on DeBakey VAD.

head (pressure difference between in- and outflow) and the flow rate at a given rotational speed of the impeller. In the case of the Debakey pump the Particle Image Velocimetry (PIV) method was used to achieve data points which define the hydraulic performance curves for rotational speeds of 7500, 8500 and 9500 rpm. First data points obtained from numerical simulations show good agreement with the experimental curves at a rotational speed of 7500 rpm where computed pressure heads at a given flow volume are within 5% of experimental values. A visualization of the pressure distribution on the surface of the pump can be seen in Fig. 2. The red colour on the top of the impeller vanes indicates high pressure due to the small gap between vanes and housing. At 9500 rpm the agreement is not so good with computed head being about 80% of the experimental one; one possible reason for this could be imperfections in the turbulence model used.

A new simulation series on the Blue Gene with an improved geometry model is now being conducted. We typically use 2048 processors, which permits to compute one pump revolution in about 3 hours. For visualization purposes, it may be necessary to store all of the computed pressure and velocity values for one revolution, corresponding to data amounts of up to 30 GB. To receive a quasi-stationary flow condition up to twenty revolutions can be required. By mesh refinement it can be confirmed if a reduction of the mesh size has no further influence on the outcome of the computations, which is a prerequisite for convergence. Former simulations were performed with a 5.5 million element mesh and the new series uses 7.5 million elements.

Blood Damage

Blood is a highly-complex fluid, a suspension of red and white blood cells (RBC, WBC) and platelets in a plasma. The volume fraction of RBC is about 45%, and blood damage, or hemolysis, can be caused by prolonged elevated stresses acting on those cells.

With increasing levels of shear stress the coin-stacked arrangement of the RBCs breaks up, the membrane starts rotating around the encapsulated fluid until, eventually, its hemoglobin contents are partly released into the plasma through small pores. This is illustrated in the upper sequence of Fig. 3. Up to a certain amount this so-called plasma-free hemoglobin can be filtered by the kidneys; higher concentrations lead to intoxication and,

in a worst-case scenario, death.

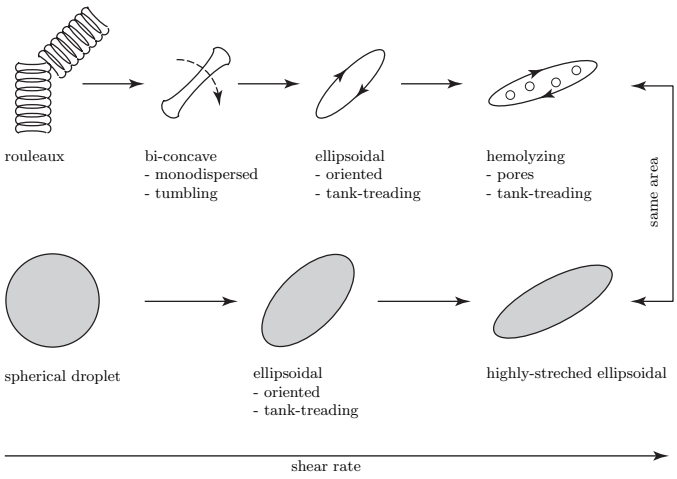


Figure 3. Behaviour under shearing: RBCs (top), droplet model (bottom).

Our group uses a novel approach to assess hemolysis. Instead of simply using exposure time and the flow-induced shear stress, the RBCs are modelled as droplets. As shown in the lower sequence in Fig. 3, hemolysis is initiated when the increase of the surface area of the droplet (which takes an ellipsoidal shape under shearing) and of a hemolyzing RBC match. Tracking the deformation along flow pathlines enables us to capture important physiological properties of the cells, such as membrane rotation (tank-treading) and relaxation in low shear regions.

Since these phenomena contribute to reduction of hemolysis, our deformation-based model showed good correspondence with experimentally-diagnosed hemolysis values[1] while the conventional stress-based models are known to overpredict hemolysis.

3 Parallel Computing

As already mentioned in Sec. 2, efficient simulations of the DeBakey VAD rely strongly on high-performance computing on powerful machines like the 8-rack Blue Gene/L system at Research Centre Jülich. The parallel implementation of XNS is based on message-passing communication libraries and exploits graph-based mesh-partitioning techniques. This means that the computational mesh is divided into subsets of elements which are then assigned to the available processors. A sample partitioning of the DeBakey blood pump is shown in Fig. 4.

After distributing the elements among the processors, a discrete form of the Navier-Stokes flow equations is solved simultaneously on all processors. While parts of the computation can be carried out within a single partition, others require communication between the processors and the exchange of data buffers. Since this inter-processor communication

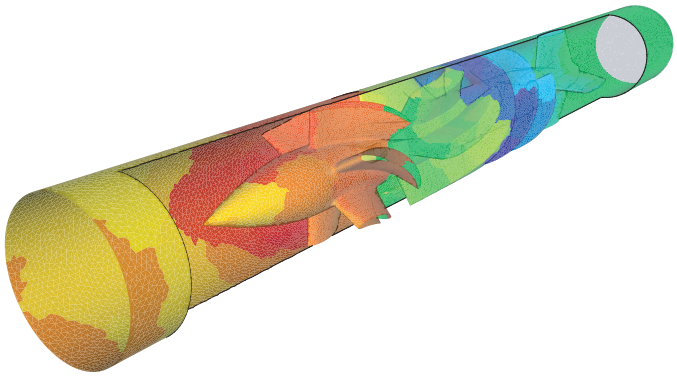
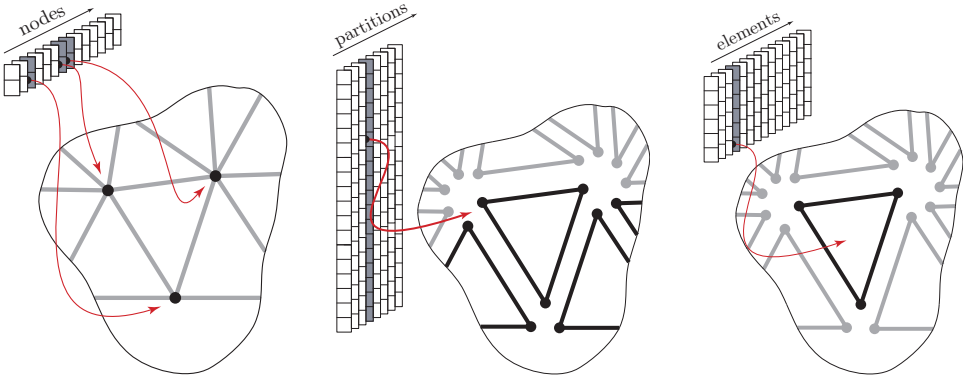


Figure 4. DeBakey blood pump: element partitions for 1024 processors.

is limited to the subdomain boundaries, the element subsets should be assigned in a way that reduces the overall boundary size to a minumum.

There are also different ways of storing the flow information during the simulation, each of which can be advantageous depending on the operation that has to be performed with the data. Fig. 5 gives an impression about the storage modes used in XNS for a triangular two-dimensional mesh. The nodal data is assigned to the processors according to the element partitioning and can be used to obtain element-based values. Moving from one level to another requires information about the element distribution as well as a connectivity matrix that specifies for each element the nodes it is composed of.



(a) node-based (b) partition-based (c) element-based

Figure 5. Storage modes for finite element data.

Performance on Jülich Blue Gene/L

In December 2006, a scaling workshop[2] for applications running on the 8-rack Blue Gene/L system in Jülich was organized and sponsored jointly by the John von Neumann Institute for Computing (NIC), IBM and the Blue Gene Consortium. Prior to the workshop, one could observe an acceptable scaling of XNS up to 1024 processors while there was no significant speed-up above that (see Fig. 6). To find the bottlenecks in the code, the communication between the processes during the simulation runs was analyzed, both with XNS internal time measurements and the SCALASCA package[3]. After improving the communication patterns of XNS, the simulation performance could be improved remarkably up to 4096 processes (see Fig. 6). A good scaling is expected also for 8192 processes; this is to be analyzed in future test runs.

The possibility to make efficient use of up to one fourth of the processors available on the Jülich Blue Gene/L is of great value for the simulations, because it allows to generate the required data in a reasonable time. Going towards the ultimate goal of the DeBakey project – the computation of an optimal design with respect to criteria mentioned in Sec. 2 – one could profit from the good scalabilty of XNS and make efficient use of even bigger machines.

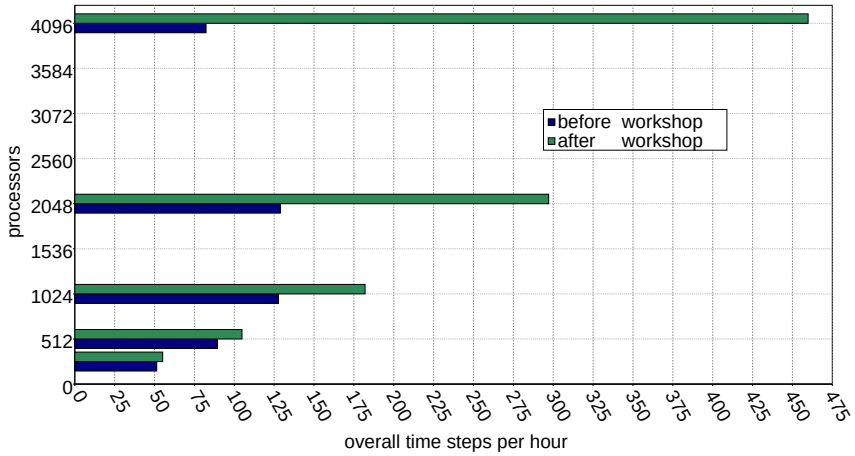


Figure 6. Performance of XNS on Jülich Blue Gene/L.

As outlined in Sec. 4, the computational demands and memory requirements in design optimization are much higher compared to plain simulations since the Navier-Stokes equations have to be solved several times on different geometries.

4 Shape Optimization for Fluids

As already mentioned, we aim to use an optimization framework that is capable of achieving an optimal design in an iterative, but automatic way, i.e., without a need for user interaction. Using such a framework, the user only needs to specify areas of the geometry, for example the impeller of the pump, that should be subjected to optimization and define

a so-called objective function to judge the quality of a design modification. This can be a difficult task: for blood pumps, an objective function that relates to hemolysis seems most reasonable, but it is important that a gradient of this function can be computed which again limits one's choice.

The state-of-the-art approach is to couple a flow solver with a grid generation tool which are both driven by an optimization algorithm. This approach has several drawbacks. Normally it is not possible to get the exact gradient in such an environment; therefore, the gradient has to be calculated with finite differences or gradient-free methods must be used. Both approaches are very costly with respect to computing time. Also, the parallelization cannot be done uniformly because the three components – mesh generator, solver, optimizer – normally use different parallelization strategies. Nevertheless, this approach can be used for some simple problems, but it is not applicable to real-world problems such as blood pumps.

We are solving the shape optimization for fluids problem within one program. We already demonstrated that this is possible for steady[4] as well as for unsteady[5] flows. In these cases we were able to show an influence of the constitutive model with respect to the optimal solution. This result showed clearly the demand for appropriate blood constitutive models as well as blood damage models. We have coupled an optimization driver with XNS, using a line search algorithm with a BFGS update method as the optimization strategy. The same approach was then used without significant changes to perform parallel optimization.

Our current work is focussed on the complex geometry deformation and new blood damage models which are more amenable to gradient computation.

Acknowledgments

The authors gratefully acknowledge the computing resources provided by the RWTH Aachen University Computing and Communication Centre, and by the John von Neumann Institute for Computing. The authors also wish to acknowledge the support the MicroMed company and of the German Science Foundation under programs GSC 111, EXC 128, SFB 540 and 401, and SPP 1253 and 1273.

References

1. D. Arora, M. Behr and M. Pasquali, *Hemolysis Estimation in a Centrifugal Blood Pump Using a Tensor-based Measure*, Artificial Organs, Vol.30 (2006): 539–547.
2. W. Frings, M. A. Hermanns, B. Mohr and B. Orth, *Report on the Jülich Blue Gene/L Scaling Workshop 2006*, Technical Report FZJ-ZAM-IB-2007-02, February 2007.
3. M. Geimer, F. Wolf, B. J. N. Wylie and B. Mohr, *Scalable Parallel Trace-Based Performance Analysis*, inSIDE Vol.4, No.2, 2006.
4. F. Abraham, M. Behr and M. Heinkenschloss, *Shape Optimization in Steady Blood Flow: A Numerical Study of Non-Newtonian Effects*, Computer Methods in Biomechanics and Biomedical Engineering, Vol.8 (2005), 127–137.
5. F. Abraham, M. Behr and M. Heinkenschloss, *Shape Optimization in Unsteady Blood Flow: A Numerical Study of Non-Newtonian Effects*, Computer Methods in Biomechanics and Biomedical Engineering, Vol.8 (2005), 201–212.